Gestió Distribuïda amb SNMP

Què farem:

- Crearem la infraestructura amb contenidors docker.
- Instal·larem i configurar el servei **snmpd** i **rsyslog** a totes les màquines.
- Configurarem el servei de registres centralitzat al server.
- Configurarem l'accés snmp des del localhost amb VACM (versió 1 o 2c)
- Configurarem l'accés snmp remot amb USM (versió 3).
- Programarem un script que vigili periòdicament si el nostre router.

1. Preparatius docker (20%)

Creeu el fitxer **dockerfile_gsx_prac5** que a partir de la imatge del laboratori anterior s'instal·lin els següent paquets: rsyslog, snmp, snmpd, snmp-mibs-downloader.
El darrer paquet està disponible a un altre repositori i caldrà afegir-lo:

```
RUN echo "deb https://deb.debian.org/debian/ bullseye non-free"
>/etc/apt/sources.list.d/non-free.list
RUN   apt update
RUN   apt-get install -y --no-install-recommends snmp-mibs-downloader
```

Un fitxer de les MIB conté alguns errors i per a evitar que ens ho notifiqui a cada consulta (tot i ser irrellevant) l'actualitzarem amb:

```
COPY SNMPv2-PDU.diff /root
RUN patch /usr/share/snmp/mibs/ietf/SNMPv2-PDU </root/SNMPv2-PDU.diff
```

Heu d'escriure un *script* anomenat **fes_docker_prac5.sh** que faci el build de la imatge, crei les 3 xarxes ISP, DMZ, INTRANET[3] i faci el run de 3 contenidors. Després de cada comanda docker de creació, abans de prosseguir, hauríeu de comprovar que l'objecte existeix, és a dir, que s'ha creat.

No caldrà aplicar els scripts de la pràctica3, cada contenidor ja tindrà una IP del rang correcte.

Per a tenir persistència de les dades en executar el contenidor **Server** afegirem un directori compartit amb l'amfitrió.

```
OPCIONS="-itd --rm --privileged"    # aplicades a tots els docker run
docker run ..... --mount type=bind,src=./practica5,dst=/root/prac5 ....
```

3 *amb els vostres esquemes d'adreçament, tal com vam fer al lab anterior*

La configuració del servei snmp serà igual a tots els contenidors. Treballarem al contenidor Server i a la resta de contenidors farem el run afegint la opció read-only: **type=bind,ro,** D'aquesta manera els canvis fets al Server estaran disponibles immediatament a tots els contenidors, on sols caldrà aplicar els canvis.

2. Configuració del registre central (20%)

Al server hi configurarem el nostre servidor syslog central.

- Modifiqueu el fitxer **/etc/rsyslog.conf** des-comentant les línies que fan que els servidor escolti el port udp 514 per a la xarxa/màscara (al final del fitxer podeu veure cap a quins fitxers s'envien els missatges segons la *priority*).
- Afegiu el fitxer **/etc/rsyslog.d/10-remot.conf** per a que els missatges rebuts dels clients es guardin a directoris separats per a cada màquina i per data.

```
# que envii els missatges de cada client a un dir diferent, separat per data
$template GuardaRemots, "/var/log/remots/%HOSTNAME%/%timegenerated:1:10:date-rfc3339%"
# No volem que el template anterior s'apliqui al localhost (el server)
:source, !isequal, "localhost" -?GuardaRemots
```

- Re-engegeu el servei i comproveu que està escoltant al port 514

~~Com els fitxers de log poden créixer molt cal anar-los rotant i comprimir-los:~~

- ~~Configureu el **logrotate** al fitxer **/etc/logrotate.d/remots** per a que roti els fitxers de **/var/log/remots/*/*** diàriament, els comprimeixi i els guardi mig any. Per això seguiu els exemples que hi ha a la seva *man page*.~~

A tots els altres contenidors caldrà dir que enviïn els missatges desitjats al servidor central:

- Editeu el fitxer **/etc/rsyslog.d/90-remot.conf** afegint que envii sols els missatges de la facility **user** al servidor remot[4]:

```
    user.*   @$IPSERVER:514
```

- Re-engegeu el servei.

Finalment, feu *l'script* **prac5_config_rsyslog.sh** el qual segons sigui el hostname el server o un altre contenidor implanti els fitxers pertinents.

4 *canvia $IPSERVER per la IP que tingui el teu contenidor server*

3. Configuració SNMP (50%)

Un cop corrent tots els contenidors anem al server i configurem els servei SNMP. Creeu un fitxer anomenat **prac5_config_snmp.sh** que faci les següents accions:

Modifiqueu els fitxers /etc/snmp/*.conf per a configurar accés per vistes (VACM).

Fitxer del servei **snmpd.conf**:[5]

- per rebre UDPs des de qualsevol màquina, comentem la línia actual del loopback i afegim:

```
agentAddress udp:161
```

- modifiqueu el **sysLocation** i el **sysContact** adientment (el vostre nom i ubicació).
- afegim la vista **vistagsx** que ens permeti veure les branques **interfaces, ip, snmp, icmp** i **ucdavis** (heu de buscar els seus OIDs)
- afegim un *community string* **cilbup** per a accés *read-only* a la vista anterior sols des del localhost (doncs la *password* viatjaria en clar).

Fitxer dels clients **snmp.conf**: posem el path de les MIB a la configuració dels agents i quins MIBs volem (tots), altrament no es poden utilitzar ni mostrar els noms dels OID en format text:

```
mibs +All
```

Per a saber els OIDs que voleu monitoritzar, podeu usar la comanda **snmptranslate** o bé, si teniu un entorn gràfic, un mib browser com ara **tkmib**. Per exemple:

```
$ snmptranslate -Td -OS UCD-SNMP-MIB::ucdavis.dskTable
```

Al final convé re-engegar el servei i comproveu que s'escolta el port 161.

Proves locals

Podeu provar les comandes de les transparències de teoria[6]. Per exemple:

```
$ snmpwalk -v 2c -c public localhost system
$ snmpwalk -v 2c -c public localhost hrSystem
```

Proveu l'accés als mibs de la Univ. California Davis:

```
$ snmptable -v 2c -c cilbup localhost UCD-SNMP-MIB::prTable
$ snmptable -v 2c -c cilbup localhost ucdavis.dskTable
$ snmptable -v 2c -c cilbup localhost ucdavis.laTable
```

5 *Vegeu els comentaris del fitxer per a veure com establir aquestes configuracions.*
6 *suggereixo almenys les de les interfícies i de la taula d'encaminament.*

Accés remot amb SNMPv3 (USM)

Per a que l'accés remot sigui més segur usarem autenticació **SHA** i xifratge **DES**.

Fitxer **snmpd.conf**:

- al principi del fitxer, creem dos usuaris, **gsxViewer** (per a l'accés de lectura) i **gsxAdmin** (per l'accés rw). El primer ha d'especificar el mecanisme d'autenticació i el segon també el d'encriptació, seguits de la *passphrase*, que respectivament hauran de ser **aut$IND** i **sec$IND** (on $IDN és el vostre DNI o equivalent del revés[7]).
- més avall definirem l'accés **rouser** i **rwuser** per als respectius usuaris, especificant a continuació el *security level*[8]. Amb aquest mètode d'accés no necessitem les vistes doncs ja és més segur[9].

Exemples de comandes de consultes amb accés amb SNMPv3:

```
snmpget -v3 -u gsxViewer -l SecurityLevel \
      -a SHA -A aut$IND $IP  system.sysDescr.0
snmpget -v3 -u gsxAdmin -l SecurityLevel \
      -a SHA -A aut$IND -x DES -X sec$IND $IP system.sysDescr.0
```

Si us convé **depurar** atureu el servei snmpd i engegueu-lo en *foreground* amb:

```
snmpd -f -V -Lod -u Debian-snmp -g Debian-snmp -I -smux -p /run/snmpd.pid
```

→ Un cop fetes les proves aturareu el procés snmpd amb un Control+C.

Una vegada comprovat que funciona bé en local, executeu aquest script a cada contenidor.

Proves SNMP remotes

Feu un script anomenat **proves_snmp_remotes.sh** que faci les proves les mateixes peticions del apartat Proves locals usant el paràmetre la IP del contenidor remot i amb accés USM.

Executeu *l'script* al server amb la IP del router i guardeu la sortida a **sortida_snmp_remota_prac5.txt**

7 *IND=$(echo $DNI | tr -cd [:digit:] | rev)*
8 *Les opcions pel security level són:* noAuthNoPriv | authNoPriv | authPriv
9 *tot i que també s'hi podrien posar:* rouser gsxViewer AuthNoPriv -V vistagsx

5. Monitoritzar el router. (10%)

Al propi router escriviu *l'script* **prac5_vigila_router.sh** el qual configuri el **cron** per a que cada **5 minuts** executi /root/**vigila_snmp.sh**. Aquest consultarà mitjançant SNMP uns quants OIDs per comprovar l'increment d'aquests entre execucions:

SNMPv2-MIB::snmpInSetRequests >0

SNMPv2-MIB::snmpInGetRequests > $LLINDAR

Quan el nombre d'aquest entre dues consultes consecutives és major que la constant $LLINDAR enviarem un missatge al syslog local avisant:

```
logger -p user.warning -t GSX "AVÍS: el valor del $OID al router ha augmentat massa: $valor ($increment)"
```

La constant $LLINDAR la fixarem a un nombre baix per a fer les proves, tot i que la versió definitiva hauria de ser alt, segons el temps entre consultes.

Proves de monitorització mínimes

- Feu un bucle que faci snmpgets cap el router suficients per a passar el llindar.
- Per a provar el set feu (usant els paràmetres -l -a -A -x -X pertinents):

```
snmpset -v3 -u gsxAdmin ... $IPROUTER ip.ipForwarding.0 = 1
snmpset -v3 ... $IPROUTER sysName.0 = 'docker container'
```

Guardeu el syslog del server corresponent al router a **log_central_router.txt**

Lliurament:

Un tgz anomenat **prac5_$COGNOMS.tgz** amb:

- dockerfile_gsx_prac5, fes_docker_prac5.sh,
- prac5_config_rsyslog.sh,
- prac5_config_snmp.sh i els fitxers de configuració,
- proves_snmp_remotes.sh, sortida_snmp_remota_prac5.txt,
- prac5_vigila_router.sh, vigila_snmp.sh, log_central_router.txt.

Referències bàsiques:

- man snmpcmd, snmpget, snmpwalk snmptable, snmptranslate
- man snmpd.conf, **snmpd.examples**
- man rsyslogd, rsyslog.conf i logrotate
- https://debian-handbook.info/browse/stable/sect.syslog.html
- Errors_SNMP_USM.pdf

Informació addicional:

www.rsyslog.com/doc/configuration/index.html

Opcional: Lectura per a saber més:

web-interfaces-for-your-syslog-server-an-overview/

Opcional: Per si algú vol provar el Cacti:

docs.cacti.net/#cacti-overview

```
docker pull gsxlabs/cacti:v2023
docker run -d --hostname nms --network=DMZ -p 8080:80 --name NMS gsxlabs/cacti:v2023
docker exec NMS /root/inici.sh

$NAVEGADORWEB http://localhost:8080/cacti
username i password: admin
```